

Memory MAM (Multi-mode Access Memory) untuk Pengolahan Citra Paralel: Prinsip, Aplikasi dan Performansi

Eril Mozef

Jurusan Teknik Elektro, Politeknik Negeri Bandung
e-mail: erilmozef@yahoo.com

Abstrak

Pada paper ini diperkenalkan sebuah desain baru memori yang kami sebut MAM (Multi-mode Access Memory). Sesuai namanya, memori ini dapat diakses dengan beberapa mode antara lain mode RAM, CAM dan Shift. Memori jenis ini telah dibuktikan memberikan manfaatnya yang besar dibidang pengolahan citra paralel terutama untuk memproses masalah citra yang bersifat lokal (misalnya: filtering, edge detection dan lain-lain) dan regional (misalnya: labeling, perhitungan luas dan keliling objek, dan lain-lain) dengan kompleksitas algoritma yang sangat baik yaitu $O(n)$ untuk citra $n \times n$ pixel. Sedangkan bila menggunakan memori RAM mencapai $O(n^2)$. Pada paper ini dibahas prinsip dasar memori MAM tersebut, aplikasi dan performansinya untuk arsitektur paralel citra berjenis 1d. Penelitian ini merupakan bagian dari proyek pengembangan arsitektur paralel citra optimal yang didanai oleh ITSF (Indonesia Toray Science Foundation).

Kata kunci : Memori asosiatif, CAM, Pengolahan citra, Pengolahan paralel, Labeling.

Abstract

This paper presents a new design of memory called MAM (Multi-mode Access Memory). According to its name, this memory can be accessed using some modes e.g. RAM, CAM, and Shift. This type of memory provides a significant benefit in the field of image processing particularly to process the local (e.g. filtering, edge detection, etc) and regional image problems (e.g. labeling, area and perimeter of objects, etc). For an image of $n \times n$ pixels, the complexities obtained were very excellent, in $O(n)$. However, these complexities could reach $O(n^2)$ when RAM memory was used. In this paper, the principle of MAM, its applications, and its performances were discussed. This research was a part of the optimal parallel architecture for image processing development project granted by ITSF (Indonesia Toray Science Foundation).

Keywords : Associative memory, CAM, Image processing, Parallel processing, Labeling.

Pendahuluan

Memori merupakan salah satu bagian terpenting pembentuk arsitektur paralel pengolahan citra. Dalam literatur terdapat berbagai jenis tipe memori antara lain yang terkenal adalah RAM (Random Access Memory) dan CAM (Content Addressable Memory). Dari survey kami terdahulu [1] dapat disimpulkan bahwa memori jenis RAM memegang peranan terhadap algoritma pengolahan citra lokal sedangkan CAM terhadap algoritma bertipe regional. Jaringan interkoneksi memegang peranan terhadap algoritma global. Sedangkan prosesor hanya memegang peranan pada algoritma tipe lokal saja.

Kebanyakan arsitektur paralel 1d (1 dimensi dengan n Processor Element) yang ada meng-

gunakan memori jenis RAM sehingga tidak memungkinkan untuk melakukan operasi asosiatif yang sangat dibutuhkan oleh algoritma berjenis regional. Sedangkan arsitektur berjenis 2d secara sifat dasarnya sudah memiliki sifat asosiatif karena topologi prosesornya yang saling berhubungan satu sama lain secara spasial.

Memori CAM dapat menyelesaikan algoritma bertipe regional dengan rata-rata kompleksitas $O(n)$. Untuk algoritma yang sama dikerjakan dengan memori RAM menghasilkan kompleksitas lebih tinggi $O(n^2)$ yang berarti lebih buruk. Atau dengan kata lain dengan CAM dapat dihasilkan kecepatan n kali lebih cepat dibandingkan dengan penggunaan RAM.

Dalam rangka merancang arsitektur paralel yang optimal, kami mencoba untuk memanfaatkan keunggulan yang diberikan oleh memori CAM untuk mengusulkan suatu memori jenis baru yang merupakan kombinasi dari beberapa fungsi antara lain RAM, CAM dan Shift. Memori ini

Catatan: Diskusi untuk makalah ini diterima sebelum tanggal 1 November 2002. Diskusi yang layak muat akan diterbitkan pada Jurnal Teknik Elektro volume 3, nomor 1, Maret 2003.

kami sebut dengan MAM (Multi-mode Access Memori).

Tujuan dari makalah ini adalah:

- memperkenalkan prinsip dan design MAM,
- membahas aplikasinya baik dari segi arsitektur maupun algoritmanya.
- menyajikan performansinya.
- menjajaki kemungkinan membangun sebuah arsitektur paralel optimal citra

Penelitian ini merupakan bagian dari proyek pengembangan arsitektur paralel optimal yang disponsori oleh ITSF (Indonesia Toray Science Foundation).

Memory CAM: State-of-the-art

Memori CAM adalah memori yang alamatannya berdasarkan isinya. Memori CAM termasuk tipe memori yang cukup tua keberadaannya dan telah banyak dijumpai dalam literatur namun manfaatnya jarang dibahas untuk arsitektur paralel pengolahan citra. Memori ini diperkenalkan pertama kali oleh Slade, dkk pada tahun 1956 [2]. Memori ini dikenal dengan nama lain misalnya memori asosiatif, memori dengan data yang dialamati, memori katalog, dll [3][4]. Pada paper ini kami mengadopsi nama yang paling sering digunakan dalam literatur yaitu CAM dan Memori asosiatif.

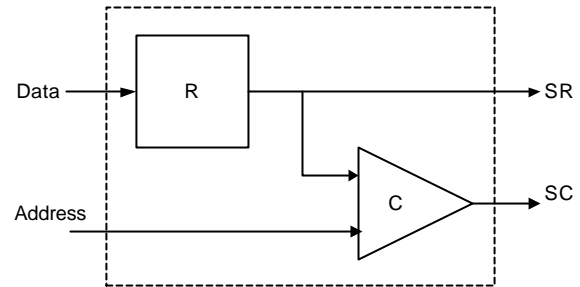
Istilah CAM banyak juga dijumpai di bidang Neural Network untuk menyatakan suatu memori yang isinya (dalam bentuk *pattern*) dapat dipanggil kembali bila pada inputnya diberikan bentuk *pattern* tersebut seperti layaknya memori otak manusia. CAM dalam konteks Pengolahan Paralel (CAM_PP) ini sama persis fungsinya dengan CAM dalam konteks Neural Network (CAM_NN). Bedanya adalah CAM_NN bersifat *massively-parallel* sedangkan CAM_PP tidak.

Keistimewaan dari memori CAM adalah dapat menggantikan m buah data dalam waktu konstan $O(1)$ atau hanya dalam sekali siklus instruksi yang mana bila dikerjakan oleh memori RAM akan memakan waktu $O(m)$.

Memori CAM memberikan manfaat yang besar dalam aplikasi praktis, misalnya Transformasi Hough [5], labeling paralel [9], penjejakan wajah [6], jaringan lokal [7], kompresi gambar [8] dan aplikasi lain yang digambarkan dalam [9][10]. Walaupun cukup banyak manfaat dan aplikasinya namun tipe ini tetap jarang dikenal

dibandingkan dengan memori RAM. Keterbatasan produk ini dipasaran membuatnya jarang dipakai dalam disain [10]. Dalam dunia arsitektur paralel tipe memori ini hanya digunakan pada pengembangan STARAN pada 1972 [11][12].

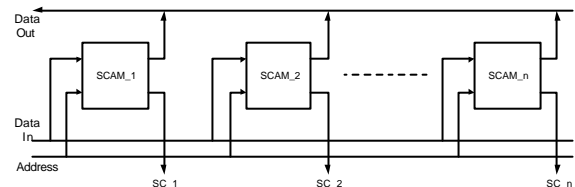
1. Sel CAM



Gambar 1. Sel CAM

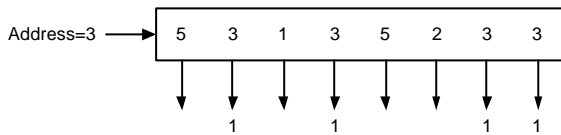
Sebuah sel CAM dibentuk dari sebuah sel memori R dan komparator C (Gambar1) Sebuah sel CAM akan memberikan jawaban/respon melalui output SC jika $SR = \text{Address}$. Output komparator ini kemudian dapat digunakan untuk merealisasikan operasi-operasi asosiatif.

2. Memori asosiatif



Gambar 2. Memori Asosiatif

Manfaat CAM terlihat jika n buah sel dibentuk secara paralel (Gambar2). Konfigurasi ini dikenal dengan nama memori asosiatif yang memungkinkan kita untuk melakukan operasi-operasi asosiatif dengan jumlah n berapapun dalam waktu konstan $O(1)$. Padahal untuk operasi yang sama memori RAM membutuhkan waktu $O(n)$. Pengalamatan memori asosiatif berarti membandingkan secara paralel isi dari setiap sel CAM dengan suatu nilai Address. Tipe pengalamatan ini dapat menghasilkan apa yang disebut “multiple-response” (Gambar 3). Dari keadaan inilah muncul beberapa jenis operasi asosiatif (lihat paragraf berikut). Keadaan ini dapat dianalogikan seperti sebuah pertanyaan dari seorang guru kepada muridnya, “Siapa yang memiliki buku ini, angkat tangan?”. Pertanyaan ini memungkinkan guru tersebut mengetahui beberapa jawaban sekaligus.

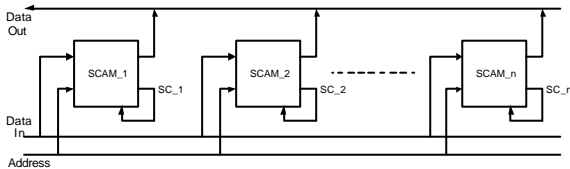


Gambar 3. Multiple-response

Bentuk umum dari memori asosiatif adalah prosesor asosiatif [4]. Prosesor asosiatif sudah pasti memiliki sifat memori asosiatif, sedangkan kebalikannya belum tentu.

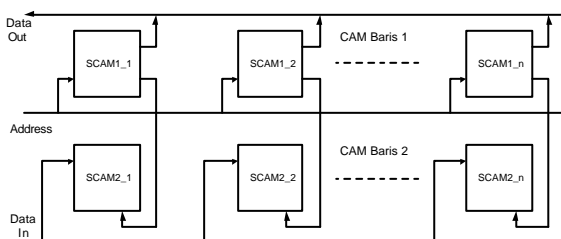
3. Operasi-operasi asosiatif

Multiple-response dari sel CAM dapat digunakan baik secara langsung maupun tidak. Untuk hal pertama, berarti kita membahas operasi determinasi jumlah sel penjawab. Sedangkan pada kasus kedua, berarti kita membahas operasi identifikasi sel penjawab [4]. Operasi-operasi ini sebenarnya penting tapi sulit direalisasi sebab pengolahan jawaban harus dilakukan secara paralel. Dalam hal yang kedua pembahasan lebih mengarah ke operasi pembacaan atau penulisan. Jika jawaban-jawaban digunakan untuk memvalidasi suatu operasi internal dalam sel untuk misalnya penulisan data, maka operasi ini disebut auto-asosiatif [13] (Gambar 4).



Gambar 4. Operasi Auto-asosiatif

Jika jawaban-jawaban digunakan untuk memvalidasi suatu operasi diluar sel, dalam hal ini untuk penulisan data maka ini disebut dengan operasi hetero-asosiatif (Gambar 5).



Gambar 5. Operasi Hetero-asosiatif

- Determinasi jumlah sel penjawab

Mengetahui jumlah sel penjawab bisa menarik untuk beberapa aplikasi. Misalnya pada aplikasi

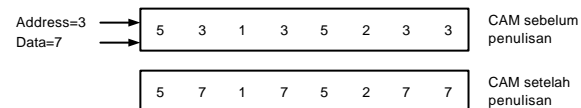
jaringan lokal dimana administrator ingin mengetahui banyaknya dan mengidentifikasi *workstation-workstation* yang aktif. Circuit yang merealisasi aplikasi ini sudah dikomersialisasikan [7].

Bentuk operasi yang paling sederhana adalah mendeteksi ada atau tidaknya sel penjawab sedangkan yang lebih kompleks adalah mengidentifikasi sel penjawab. Identifikasi ini dapat digunakan untuk *sorting* data.

- Identifikasi sel penjawab

Bentuk operasi ini adalah menyeleksi satu sel diantara sel-sel penjawab. Penyeleksian bisa dilakukan baik secara acak maupun secara prioritas. Pendekatan yang paling mudah adalah melakukan operasi penggeseran dan mendeteksi jawaban pertama yang muncul.

- Operasi penulisan CAM pada mode auto-asosiatif



Gambar 6. Penulisan CAM Auto-asosiatif

Definisi penulisan CAM pada mode auto-asosiatif dari suatu Data D dengan Address A, misalnya, adalah pencarian secara paralel semua sel berisi A dan meng-updatenya dengan data D. Operasi ini dikenal pula dengan sebutan PSMU (Parallel Search and Multiple Update) [9]. Operasi ini digambarkan dengan algoritma berikut:

```

For I=0 to n-1 do (in paralel),
  if CAM[i]==AlamatCAM
  then CAM[i]=DataCAM
  end_if
end_for

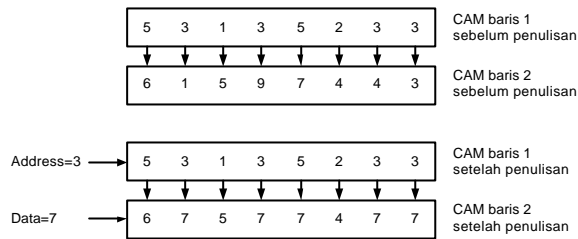
```

Operasi penulisan CAM pada mode auto-asosiatif diilustrasikan dengan Gaambar 6.

- Operasi penulisan CAM pada mode hetero-asosiatif

Operasi penulisan CAM pada mode Hetero-asosiatif ini diilustrasikan pada Gambar 7. Misalnya ada dua baris CAM berisi 8 sel dengan masing-masing sebuah harga. Definisi penulisan CAM pada mode hetero-asosiatif dari Data=7

dengan Address=3 adalah pencarian secara paralel pada CAM baris 1, seluruh sel berisi harga 3 dan menggantikannya dengan harga 7 seluruh CAM baris 2 yang berindeks sama.

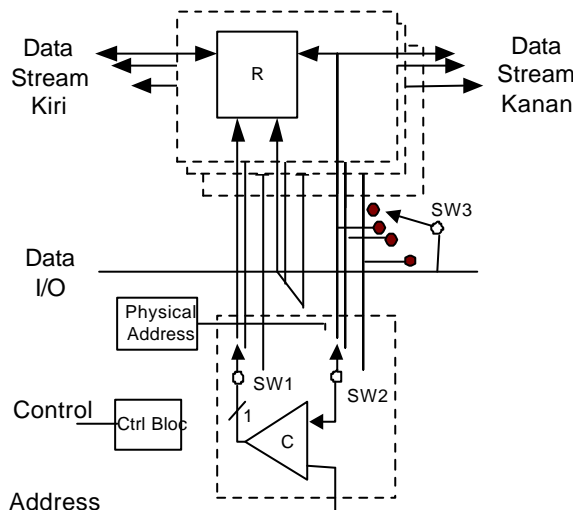


Gambar 7. Penulisan CAM Hetero-asosiatif

• Operasi pembacaan CAM

Operasi pembacaan data CAM secara umum tidak diimplementasikan sebab secara definisi, alamat adalah sama dengan data itu sendiri.

Memory MAM: Prinsip

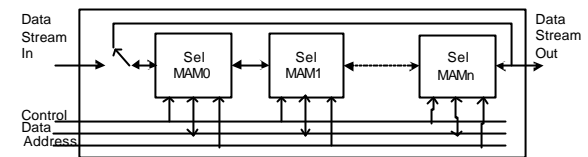


Gambar 8. Sebuah sel MAM

Secara prinsip, memori MAM merupakan gabungan dari 3 buah fungsi akses: RAM, CAM dan Shift. Sebuah sel MAM memiliki m buah register R dan sebuah komparator C (Gambar 8). Register R merupakan register B bit yang dapat menyimpan data dan menggesernya ke kiri atau ke kanan. Komparator digunakan untuk mengamati sel dengan cara membandingkan Address (data eksternal) dengan data internal melalui switch SW2. Hasil perbandingan digunakan untuk mengaktifkan data menuju ke dalam R ataupun keluar R melalui SW1. SW3 digunakan untuk memilih data yang akan dikeluarkan ke port Data I/O dari R.

1. Bank MAM

Beberapa buah sel MAM yang diparalel dapat membentuk sebuah bank MAM yang terdiri dari n sel untuk citra berukuran $n \times n$ (Gambar 9). Bank MAM ini memiliki port Control, Data (in/out), Address, Data Stream In dan Data Stream Out. Dua port terakhir ini dipergunakan untuk memasukkan dan mengeluarkan data video secara cepat dengan mode FIFO. Input/Output data stream ini dapat bersifat internal dan eksternal yang diatur melalui switch internal.



Gambar 9. N sel MAM membentuk sebuah bank MAM

2. Fungsi-fungsi MAM

Sebuah bank MAM memiliki 13 instruksi yang dapat dilihat pada Tabel 1. Selected-row Address dipergunakan untuk memilih baris mana pada bank yang akan dibandingkan. Modified-row Address dipergunakan untuk memilih baris mana yang akan dimodifikasi. RAM/CAM address dipergunakan untuk memilih suatu lokasi (bila RAM mode diaktifkan) dan memilih suatu isi (bila CAM mode diaktifkan). Data I/O dipergunakan untuk menentukan nilai data yang akan diberikan atau nilai data yang akan diterima.

Untuk mode CAM-Write, bila Selected-Row_Address=Modified-Row_Address maka operasi ini dikenal dengan auto-asosiatif. Bila tidak maka dikenal dengan operasi hetero-asosiatif. CAM-Write ada 2 jenis: CAM-Write-Ext-data dan CAM-Write-Int-data. Pada CAM-Write-Ext-data, data penggantinya berasal dari eksternal dalam hal ini dari Data I/O. Sedangkan pada CAM-Write-Int-data, data penggantinya berasal dari internal dalam hal ini dari Register yang dipilih.

Instruksi Shift digunakan untuk menggeser data ke kiri atau ke kanan. Sedangkan instruksi Rotate digunakan untuk memutar data ke kanan atau ke kiri. Sebuah instruksi yang sangat bermanfaat adalah instruksi Frame-Capture. Instruksi ini dipergunakan untuk mengambil satu frame pada stream video 30 frame per detik.

Tabel 1: Instruksi bank MAM

	Control	Address			Data I/O	
	Operation mode (4 bit)	Selected-row Address (m bit)	Modified-row Address (m bit)	RAM/CAM Address (2logn bit)	Data In (2logn bit)	Data Out (2logn bit)
1	RAM-Write	Row-selected	X	Indeks lokasi	Data In	X
2	RAM-Read	Row-selected	X	Indeks lokasi	X	Data Out
3	CAM-Write-Ext-data-‘=’	Row-selected	Row-modified	Indeks Isi	Data In	X
4	CAM-Write-Ext-data-‘>’	Row-selected	Row-modified	Indeks Isi	Data In	X
5	CAM-Write-Ext-data-‘<’	Row-selected	Row-modified	Indeks Isi	Data In	X
6	CAM-Write-Int-data-‘=’	Row-selected	Row-modified	Indeks Isi	X	X
7	CAM-Write-Int-data-‘>’	Row-selected	Row-modified	Indeks Isi	X	X
8	CAM-Write-Int-data-‘<’	Row-selected	Row-modified	Indeks Isi	X	X
9	Shift-Right	Row-selected	X	X	X	X
10	Shift-Left	Row-selected	X	X	X	X
11	Rotate-Right	Row-selected	X	X	X	X
12	Rotate-Left	Row-selected	X	X	X	X
13	Frame-capture	Row-selected	X	X	X	X

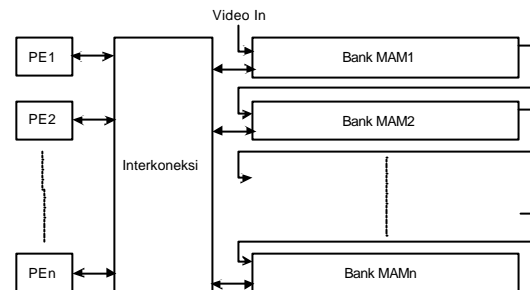
Karena total instruksi sebanyak 13 maka jumlah bit untuk memilih instruksi tersebut adalah sebesar 4 bit (konstan). Untuk Selected-Row dan Modified-Row Address adalah sebesar m bit. Sedangkan untuk RAM/CAM Address dan Data I/O adalah sebesar $2\log n$ bit (untuk citra $n \times n$).

Aplikasi MAM pada Arsitektur dan Algoritma Paralel

Sebuah model arsitektur paralel 1d dapat dibangun dari n bank MAM, n PE (Processor Element) dan jaringan interkoneksi (Gambar 10). Model ini adalah generalisasi dari arsitektur spesifik untuk proses labeling citra yang telah kami kembangkan terdahulu [14][15][16][17].

Secara umum model ini dapat dijelaskan sebagai berikut. Sebuah bank MAM apapun melalui jaringan interkoneksi dapat diakses oleh sembarang PE. Begitupun sejumlah k bank MAM ($0 < k \leq n$) dapat diakses oleh sejumlah k PE ($0 < k \leq n$). Pada mode tulis, satu atau beberapa bank dapat ditulis oleh sebuah PE. Pada mode baca, satu atau beberapa PE dapat membaca sebuah bank MAM.

Algoritma citra lokal dan regional telah berhasil dipecahkan dengan MAM secara efisien dan optimal. Algoritma lokal misalnya Filtering dan Edge detection. Sedangkan algoritma regional misalnya, perhitungan luas dan keliling objek. Untuk lebih jelasnya algoritma tersebut dapat mengacu pada [19][20][21][22].



Gambar 10. Arsitektur paralel 1d dapat dibangun dengan N buah bank MAM

Hasil dan Performansi

Hasil yang didapat dengan menggunakan model arsitektur yang telah didefinisikan pada paragraf sebelumnya diberikan pada Tabel 2. Tabel ini menunjukkan perbandingan kompleksitas algoritma yang dicapai dengan arsitektur jenis lain. Tabel ini mengacu pada hasil survey yang telah kami lakukan sebelumnya [1]. Disini kita lihat bahwa dengan MAM, untuk algoritma lokal dan regional didapatkan kompleksitas optimal $O(n)$. Sedangkan untuk algoritma bertipe global berhasil mendapatkan kompleksitas yang relatif kecil $O(n \log n)$, namun masih belum optimal.

Proses labeling objek yang pernah diujicobakan dengan menggunakan memori MAM yang diimplementasikan pada sebuah arsitektur paralel *dedicated* [16] menunjukkan hasil yang sangat mengagumkan yaitu 0,416 ms jauh mengalahkan lawan-lawannya sesama arsitektur 1d (Tabel 3). Hasil ini dicapai dengan hanya menggunakan 256 PE untuk citra 512×512 dan clock sebesar 100 ns. Satu-satunya yang dapat menandingi adalah arsitektur IUA namun dengan 262144 PE!.

Tabel 2. Perbandingan performansi arsitektur paralel yang ada

Algoritma	Sequensial, P=O(1)	1d, P=O(n), Bus lokal, RAM		1d, P=O(n), Bus reconfi gurable, RAM		1d, P=O(n), orthogonal, RAM		1d, P=O(n), Tree structure, MAM		2d, P=O(n ²), Bus reconfi gurable, CAM		2d, P=O(n ²), Bus global, CAM		3d, P=O(n ³), Bus reconfi gurable, CAM	
	T	T	P O	T	P O	T	P O	T	P O	T	P O	T	P O	T	P O
Filtering (Lokal)	O(n ²)	O(n)	Y	O(n)	Y	O(n)	Y	O(n)	Y	O(1)	Y	O(1)	Y	O(1)	N
Edge det. (Lokal)	O(n ²)	O(n)	Y	O(n)	Y	O(n)	Y	O(n)	Y	O(1)	Y	O(1)	Y	O(1)	N
Histogram ming (Global)	O(n ²)	O(nlogn) G≤n	N	O(n) G≤n	Y	O(n) G≤n	Y	O(nlogn) G≤n	N	O(√n) G=n	N	O(√n) G=n	N	O(1)	N
90° rotation (Global)	O(n ²)	O(n ²)	N	O(n)	Y	O(n)	Y	O(nlogn)	N	O(n)	N	O(n)	N	O(1)	N
180° rotation (Global)	O(n ²)	O(n ²)	N	O(n)	Y	O(n)	Y	O(nlogn)	N	O(n)	N	O(n)	N	O(1)	N
Labeling (Regional)	O(n ^{2*})	O(n ²)	N	O(n ²)	N	O(n ²)	N	O(n)	Y	O(log n)	N	O(log n)	N	O(1)	N
Perimeter (Regional)	O(n ²)	O(Cn)	N	O(Cn)	N	O(Cn)	N	O(n)	Y	O(n)	N	O(n)	N	O(1)	N
Area (Regional)	O(n ²)	O(Cn)	N	O(Cn)	N	O(Cn)	N	O(n)	Y	O(n)	N	O(n)	N	O(1)	N

Hasil simulasi yang telah kami lakukan untuk memori MAM ini dapat dilihat pada [18].

Tabel 3. Perbandingan kecepatan arsitektur dengan memori MAM dan arsitektur paralel lainnya

Arsitektur	Jumlah Prosesor	Ukuran Citra (pixel)	Clock (ns)	Waktu proses labeling (ms)
Encore Multimax	20	512x512		22700
Butterfly	100+	512x512		7200
Non-Von Supercomputer		512x512		1000
Connection Machine		512x512		400
Hierarchical Bus Architecture	100+	512x512		370
MPP	16384	128x128	100	94,6
WARP		512x512		75
Cube	256	512x512		14
Systolic	512	512x512	25	13,5
Mosaic	16384	512x512		6
Sympati	32	512x512		4,9/scanning
GFLOPS	256	256x256	100	3,5
Arsitektur dengan MAM	256	512x512	100	0,416
IUA	262144	512x512		0,005

Kesimpulan

Pada paper ini telah diperkenalkan prinsip dan disain sebuah memori baru MAM (Multi-mode Access Memori). Memori ini telah menunjukkan manfaatnya yang sangat besar untuk bidang pengolahan citra paralel. Memori ini memberikan hasil yang sangat signifikan dalam hal penyelesaian algoritma-algoritma citra bersifat regional dan asosiatif dengan kompleksitas $O(n)$ yang mana dengan memori RAM konvensional dikerjakan dengan kompleksitas $O(n^2)$. Begitupun untuk algoritma bertipe lokal, MAM dapat memberikan kompleksitas $O(n)$ yang mana tipe algoritma ini tidak dapat diselesaikan hanya dengan memori CAM. Dari 3 jenis algoritma pengolahan citra (lokal, regional dan global)

hanya algoritma global saja yang belum sanggup diselesaikan dalam waktu $O(n)$ oleh MAM. Untuk mendapatkan struktur paralel yang optimal, ketiga jenis algoritma harus mendapatkan kompleksitas algoritma $O(n)$. Sehingga penelitian kami berikutnya diarahkan pada pengembangan jaringan interkoneksinya.

Ucapan Terimakasih

Proyek ini dapat terealisasi berkat bantuan Grant dari ITSF (Indonesia Toray Science Foundation) untuk program tahun 2000. Penulis mengucapkan terimakasih banyak atas bantuannya tersebut.

Daftar Pustaka

- [1] E. Mozeff, Arsitektur Paralel Pengolahan Citra dan Performansi Optimal, Prosiding Ilmu Komputer dan Teknologi Informasi III (SNKK3), Agustus 2002, Jakarta, pp. 39-44.
- [2] A.E. Slade and H.O. McMahon, A Cryotron Catalog Memory System, Proc. 1956 East Joint Computer Conferences, 1956, pp. 115-120.
- [3] J. Minker, An Overview of Associative or Content-Addressable Memory Systems and a KWIC Index to the Literature: 1956-1970, Computing Reviews, Oct, 1971, pp. 453-504.
- [4] B. Parhami, Associative Memories and Processors: An overview and selected bibliography, Proc. of IEEE, vol. 61, no. 6, June, 1973, pp. 722-730.
- [5] M. Nakanishi and T. Ogura, A real-time CAM-based Hough transform algorithm and

- its performance evaluation, Proc. of ICPR 96, 1996, pp. 516-521.
- [6] Y. Fujino, T. Ogura, and T. Tsuchiya, Facial image tracking system architecture utilizing real-time labeling, Proc. SPIE-Int. Soc. Opt. Eng., vol. 2094, no. Pt.1, 1993, pp.2-11.
- [7] Am99C10A, 256x48 Content Addressable Memory, Advanced Micro Devices, Sunnyvale, CA, 1990.
- [8] S. Panchanathan and M. Goldberg, Vector-centered CAM architecture for image coding using vector quantization, SPIE Visual Communications and Image processing, vol. 1199, 1989, pp. 1084-1094.
- [9] Y.C. Shin, R. Sridhar, V. Demjaneko, P.W. Palumbo, and S.N. Srihari, A special purpose Content-Addressable Memories chip for real-time image processing, IEEE Journal of Solid-State Circuits, vol. 27, no. 5, Mai, 1992, pp. 737-744.
- [10] L. Chisvin and R.J. Duckworth, Content-Addressable and assosiative memory: alternatives to the ubiquitous RAM, Computer, July, 1989, pp. 51-64.
- [11] W.E. Snyder, and C.D. Savage, Content-Addressable read/write memories for image analysis, IEEE Transactions on Computers, vol. C-31, no. 10, Oct., 1982, pp. 963-968.
- [12] K.E. Batcher, STARAN parallel processor system hardware, Proc. of the National Computer Conf., vol.43, 1974, pp. 405-410.
- [13] T. Kohonen, Content-Addressable Memories, Springer-Verlag (Editors: King Sun Fu), 1980
- [14] E. Mozef, S. Weber, J. Jaber, and E. Tisserand, Real-time connected component labeling on one-dimensional array processors based on Content-Addressable Memory: optimization and implementation, UMIST-IEEE 3rd International Workshop on Image and Signal Processing, Manchester, United Kingdom, Nov. 96, pp. 691-694.
- [15] E. Mozef, S. Weber, J. Jaber, and E. Tisserand, Parallel architecture dedicated to connected component analysis, IAPR-IEEE 13th International Conference on Pattern Recognition, Vienna, Austria, August 96, pp. 699-703. (IEEE Computer Society Press).
- [16] E. Mozef, S. Weber, J. Jaber, and G. Prieur, Parallel architecture dedicated to connected component labelling in $O(n \log n)$: FPGA Implementation, SPIE International Symposium on Las., Opt., and Vision for Product. In Manufact. II, Micropolis, Besancon, France, June 96, pp. 120-125.
- [17] E. Mozef, S. Weber, J. Jaber, and E. Tisserand, Architecture dediee a l'algorithme parallel $O(n \log n)$ d'etiquetage de composantes connexes, 3eme Journee Adequation Algorithme Architecture en Traitement du Signal et Images, Toulouse, France, Jan. 96, pp. 83-89. (In collaboration with IEEE signal processing).
- [18] I. Darmawan, W.T. Hartono, E. Mozef, S. Sutikno, Kuspriyanto, VHDL Design and Simulation of MAM Memori for LAPCAM Parallel Architecture for Image Processing, IEEE Inter. Conf. on Inteligent Computing Systems APCCAS, Bali, October, 2002, Accepted.
- [19] E. Mozef, S. Weber, J. Jaber, and C. Bataille, LAPCAM, Linear Array of Processors using Content-Addressable Memories: A new design of machine vision for parallel image computations, IAPR International Workshop on Machine Vision Applications, Tokyo, Japan, Nov. 96, pp. 166-169.
- [20] E. Mozef, S. Weber, J. Jaber, and E. Tisserand, Design of linear array processors with Content-Addressable Memory for intermediate level vision, ISCA-IEEE 9th International Conference on Parallel and Distributed Computing Systems, Dijon, France, Sept. 96, pp. 585-588.
- [21] E. Mozef, Conception et etude d'une architecture parallel a reseau lineaire de processeurs et memoire CAM pour le traitement d'image, Disertation, University of Nancy, 1997.
- [22] H. Rabah, H. Mathias, E. Mozef, D. Torres, Linear Array Processors with Multiple Access Mode Memory for Real-Time Image Processing, IEEE Inter. Conf. on Inteligent Computing Systems APCCAS, Bali, October, 2002, Accepted.